

IN THE CLAIMS:

B1
1. (Currently amended) A method of implementing a switch instruction in an IA64 architecture based data processing device, comprising:

receiving a call to the switch instruction, the call including one or more parameters for the switch instruction, wherein the one or more parameters includes a range of branch address, the range being defined by a high value and a low value;

loading a plurality of predicate registers with values associated with a plurality of branch addresses based on the one or more parameters; [[and]]

calling an instruction associated with one of the plurality of branch addresses based on the values of the plurality of predicate registers;

determining if the low value is lower than a lowpredicate;

setting a first register value to $2^{**}(\text{lowpredicate} - \text{low value})$ if the low value is lower than the lowpredicate; and

setting the first register value to $2^{**}(\text{lowpredicate})$ if the low value is not zero, where lowpredicate is a predicate register number of a lowest numbered predicate register.

2. (Canceled)

2/
3. (Currently amended) The method of claim [[2]] 1, wherein the one or more parameters includes an index into the range of branch addresses, the method further comprising:

setting a second register value equal to the index if the low value is lower than the lowpredicate; and

setting the second register value equal to a difference between the index and the low value if the low value is not lower than the lowpredicate.

34/
4. (Original) The method of claim ²3, further comprising shifting the first register value to the left by the second register value.

131
4
3
5. (Original) The method of claim 4, further comprising moving the first register value to the plurality of predicate registers.

5
6. (Original) The method of claim 5, wherein moving the first register value to the plurality of predicate registers is performed using a mask identifying predicate registers to be loaded.

6
7. (Original) The method of claim 1, wherein the switch instruction is a Java tableswitch instruction.

7
8. (Original) The method of claim 1, wherein the method is implemented in a Java Virtual Machine.

8
9. (Original) The method of claim 1, wherein the switch instruction is a dense switch statement in C.

9
10. (Original) The method of claim 3, wherein the one or more parameters includes a default address, and wherein if the first register value is shifted outside of a range of the plurality of predicate registers, an instruction associated with the default address is called.

10
11. (Currently amended) An apparatus for executing a switch instruction in an IA64 architecture based data processing device, comprising:

means for receiving a call to the switch instruction, the call including one or more parameters for the switch instruction, wherein the one or more parameters includes a range of branch address, the range being defined by a high value and a low value;

means for loading a plurality of predicate registers with values associated with a plurality of branch addresses based on the one or more parameters; [[and]]

means for calling an instruction associated with one of the plurality of branch addresses based on the values of the plurality of predicate registers;

means for determining if the low value is lower than a low predicate;

means for setting a first register value to $2^{**}(\text{lowpredicate-low value})$ if the low value is lower than the lowpredicate; and

means for setting the first register value to $2^{**}(\text{lowpredicate})$ if the low value is not lower than the lowpredicate, where lowpredicate is a predicate register number of a lowest numbered predicate register.

12. (Canceled)

13. (Currently amended) The apparatus of claim ¹⁰11, wherein the one or more parameters includes an index into the range of branch addresses, the apparatus further comprising:

means for setting a second register value equal to the index if the low value is lower than the lowpredicate; and

means for setting the second register value equal to a difference between the index and the low value if the low value is not lower than the lowpredicate.

¹²14. (Original) The apparatus of claim ¹¹13, further comprising means for shifting the first register value to the left by the second register value.

¹³15. (Original) The apparatus of claim ¹²14, further comprising means for moving the first register value to the plurality of predicate registers.

¹⁴16. (Original) The apparatus of claim ¹³15, wherein the means for moving the first register value to the plurality of predicate registers uses a mask to identify predicate registers to be loaded.

¹⁵17. (Original) The apparatus of claim ¹⁰11, wherein the switch instruction is a Java tableswitch instruction.

¹⁶18. (Original) The apparatus of claim ¹⁰11, wherein the switch instruction is a dense switch statement in C.

17
19. (Original) The apparatus of claim 14¹², wherein the one or more parameters includes a default address, and wherein if the first register value is shifted outside of a range of the plurality of predicate registers, an instruction associated with the default address is called.

B1
18
20. (Currently amended) A computer program product in a computer readable medium for executing a switch instruction in an IA64 architecture based data processing device, comprising:

first instructions for receiving a call to the switch instruction, the call including one or more parameters for the switch instruction, wherein the one or more parameters includes a range of branch address, the range being defined by a high value and a low value;

second instructions for loading a plurality of predicate registers with values associated with a plurality of branch addresses based on the one or more parameters; [[and]]

third instructions for calling an instruction associated with one of the plurality of branch addresses based on the values of the plurality of predicate registers;

fourth instructions for determining if the low value is lower than a lowpredicate;

fifth instructions for setting a first register value to $2^{**}(\text{lowpredicate} - \text{low value})$ if the low value is lower than the lowpredicate; and

sixth instructions for setting the first register value to $2^{**}(\text{lowpredicate})$ if the low value is not lower than the lowpredicate, where lowpredicate is a predicate register number of a lowest numbered predicate register.

21. (Canceled)

19
22. (Currently amended) The computer program product of claim 20¹⁸, wherein the one or more parameters includes an index into the range of branch addresses, the computer program product further comprising:

seventh instructions for setting a second register value equal to the index if the low value is lower than the lowpredicate; and

Eighth instructions for setting the second register value equal to a difference between the index and the low value if the low value is not lower than the lowpredicate.

131 ²⁰~~28~~. (Original) The computer program product of claim ¹⁹~~22~~, further comprising ninth instructions for shifting the first register value to the left by the second register value.

²¹~~24~~. (Original) The computer program product of claim ²⁰~~23~~, further comprising tenth instructions for moving the first register value to the plurality of predicate registers.

²²~~25~~. (Original) The computer program product of claim ²¹~~24~~, wherein the tenth instructions for moving the first register value to the plurality of predicate registers are executed using a mask identifying predicate registers to be loaded.

²³~~26~~. (Original) The computer program product of claim ¹⁸~~20~~, wherein the switch instruction is a Java tableswitch instruction.

²⁴~~27~~. (Original) The computer program product of claim ¹⁸~~20~~, wherein the computer program product is executed in a Java Virtual Machine.

²⁵~~28~~. (Original) The computer program product of claim ¹⁸~~20~~, wherein the switch instruction is a dense switch statement in C.

²⁶~~29~~. (Original) The computer program product of claim ²⁰~~23~~, wherein the one or more parameters includes a default address, and wherein if the first register value is shifted outside of a range of the plurality of predicate registers, an instruction associated with the default address is called.